

# Automated Modelica Package Generation of Parameterized Multibody Systems in CATIA

Daniel Baumgartner, Andreas Pfeiffer

German Aerospace Center (DLR), Institute of System Dynamics and Control  
82234 Wessling, Germany

[Daniel.Baumgartner@dlr.de](mailto:Daniel.Baumgartner@dlr.de), [Andreas.Pfeiffer@dlr.de](mailto:Andreas.Pfeiffer@dlr.de)

## Abstract

In early stages of the product development process computer-aided design (CAD) and multibody simulation (MBS) work concurrently to build a virtual mechanical system. While CAD handles the geometric design and space analysis, MBS leads to a deeper understanding of the dynamic behavior of the future system. The CAD system has to provide physical and geometrical data, such as mass, inertia and connecting frames in order to improve simulation results. Automation at this point helps to create consistent simulation and design models and shortens the amount of time needed to produce realistic simulation results. Based on Visual Basic for Applications (VBA) a method is implemented to automatically generate an isolated Modelica model package or a single Modelica model from CATIA assemblies or parts. The introduction of design controlling parameter variables in addition to the multibody data enables optimization loops between multibody simulation and the related CAD model. An example demonstrates the three main steps of the presented method, divided into model processing, package generation and parameterized package update. Furthermore, the update process is integrated in a manual parameter variation as well as an automated optimization routine to enable parametric design studies coupled with multibody simulation.

**Keywords:** *CATIA, Modelica, Dymola, Multibody Simulation, Parameterized Models, Package Generation, Optimization*

dimensional representation of virtual assemblies and subordinate parts. During the development process both engineering disciplines need to exchange data in the loop. Figure 1 clarifies the relations between the two domains. Physical and geometrical data is extracted from CAD and used as input for MBS. Dynamic simulation and optimization proposes changes in the design. Traditionally both domains have independent experts for the applied software. The generated data is shared manually between them over agreed-upon fixed interfaces. Automating this time consuming and error-prone repetitive procedure is a step toward the improvement of the product development process. The automation approach depends on the used design and simulation software. In the context of the presented work Modelica [1] is used as modeling language for multibody simulation and CATIA [2] as design software.

Existing methods for automated translation from CATIA to Modelica take different boundary constraints into account. The approach in [3] is based on a kinematic skeleton in Modelica. This Modelica file is imported in CATIA and the user maps CATIA data to the kinematic skeleton through a graphical user interface. Finally the multibody data is exported to the Modelica file. In this approach the design and the simulation work is divided and executed by different persons. The design process is the master. [4] presents a method for kinematic coupling of CATIA models with Modelica. The basic requirement for this method is a working kinematic structure in CATIA. The kinematic structure from the CATIA

## 1 Introduction

Multibody simulation and computer-aided design are gaining significantly in importance during the virtual product development process. Typical tasks of multibody analysis include the computation of dynamic behavior, coupling forces or modal analysis. CAD systems on the contrary provide a three-

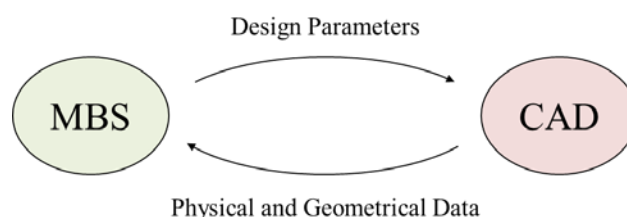


Figure 1: Relation between MBS and CAD

kinematics workbench is translated to Modelica Code with attention to kinematic loops and special treatment of redundant joints.

In some cases, however, this time-consuming setup of a kinematic structure in CATIA is not desired or not possible due to open kinematic chains. In terms of dynamic preinvestigation at early stages of the product development process, however, it can be helpful to focus on the simulation process as master in order to generate fast results. Therefore a workflow is presented here, which matches the aforementioned requirements. An isolated package or a model is generated from CATIA assemblies and parts to be used for simulation and optimization in a Modelica simulation environment, based on the Dymola framework. The mandatory model preparation work can be executed solitarily by a simulation expert with basic CATIA experience.

The upcoming sections of this paper are organized as follows. Section 2 gives a detailed overview of the used software and presents the classic manual approach of virtual development. The following section presents an automated translation approach divided into model processing, package generation, parameter handling and package updating. A bicycle rear suspension serves as example and demonstrates the usage of this method in Section 4. Section 5 summarizes the main conclusions together with an outlook to future work.

## 2 Multibody Simulation and Computer-Aided Design

This section presents the tools used in the context of the paper. Firstly, multibody simulation in Modelica with Dymola is described. The following subsection focusses on the design process in CATIA. Subsection 2.3 finally describes the classic workflow of product development between the presented domains.

### 2.1 Multibody Simulation in Modelica/Dymola

In general, multibody models consist of rigid bodies and joints. The bodies are modeled as a skeleton of fixed translations  $r_{0n}^0$  and rotations  $T_{0n}$  between connecting frames together with body mass and inertias representing the real products shape (Figure 2). Joints between the components allow certain degrees of freedom and determine the kinematic behavior of the mechanical system. The Modelica Standard Library provides algebraic and dynamic models for multibody systems [5]. These can be combined

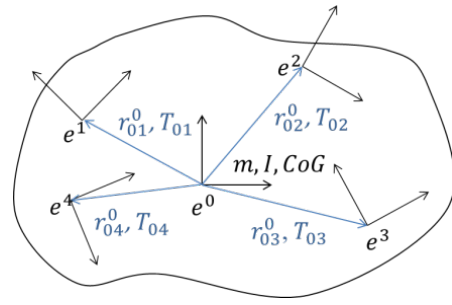


Figure 2: Geometric skeleton of a rigid body in a multibody simulation

graphically in Dymola to create multibody systems for dynamic model simulations. Based on the equilibrium of forces Differential Algebraic Equations (DAE) are formulated and solved. The outputs of the simulation experiments are motion, forces and modal behavior of the considered system. Model parameters can be adapted in optimization loops to enhance the system dynamics to match the desired behaviors.

### 2.2 Computer-Aided Design in CATIA

The design process in CATIA is mainly carried out by using the Part Design and the Assembly Design Workbench. The Part Design Workbench is used for modeling of single parts made from basic geometric elements such as points, lines and planes. These elements are combined to create two-dimensional sketches. Elementary operations transform these sketches to surfaces and volumes. Parameterizing variables, in the following referred to as design parameters, can be defined on top-level to create different representations of a part. In order to determine the part's mass and inertia, its density has to be defined through the assignment of materials.

The Assembly Design Workbench, on the contrary, is used to design mechanical systems from existing parts or sub products. Geometrical constraints, like coincidences and offsets, represent the locked degrees of freedom by the connecting joints between

Data Type	Parameters	Workflow
Physical Data	Mass, Inertia, Center of Gravity (CoG)	CAD to MBS
Geometrical Data	Connecting Frames, Points of Interest	CAD to MBS
Visualization	Shape representation	CAD to MBS
Design Parameters	Lengths, Angles	MBS to CAD

Table 1: Workflow between MBS and CAD

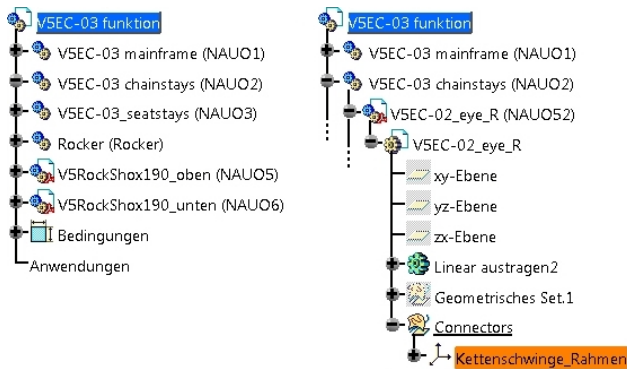


Figure 3: Top-level and connector level structure of an assembly in CATIA

the components. These relations allow manipulation according to the system's degrees of freedom for space and collision analysis. As previously mentioned multibody data is needed as input for the simulation work. There are different types of data: physical information, geometry, visualization and design parameters. Table 1 shows the workflow between MBS and CAD.

CATIA is organized in a hierarchic model structure. Figure 3 shows the top-level products and the nested structure on connector level. Each component of an assembly can consist of further products and parts on various levels. In terms of kinematic investigations it is sufficient to focus on the top-level of the product structure where the moving parts of a mechanism are located. Therefore the physical and geometrical data of the sub products is transformed to the first level of the assembly structure. Mass, inertia and center of gravity is computed for all sub products individually and merged to a single representative data set.

### 2.3 Classic Workflow

Traditionally the workflow can be described as follows. After the conceptual phase of the product design the kinematic structure is defined and initial modeling work in MBS and CAD is triggered. Important interfaces between both domains are fixed in order to establish consistent data exchange. Based on the early design model multibody data is extracted by the CAD expert and passed to the simulation expert. This data is used by the MBS expert to update the related simulation models. Accordingly, the MBS expert analyzes the dynamic model behavior and suggests changes to the design to match predefined objectives. In the next step these changes are applied to the design model and the multibody data is updated. These elemental tasks are repeated in the loop

until the dynamic behavior of the system meets the desired requirements.

Automating the repetitive data extraction from CAD to MBS helps to create consistent data and shortens the amount of time needed to produce realistic simulation results. As already mentioned, different approaches exist to improve the mutual work of design and simulation experts. At early stages of the product development process it can be more helpful to focus on the simulation process as master to rapidly generate simulation results. The presented approach in the next section handles the kinematic structure exclusively on the simulation side.

## 3 Automated Modelica Package Generation

The main contribution of the presented work is a method for an automated Modelica package generation with specific characteristics. The introduction of design parameters in addition to the multibody data allows parameter variations with MBS and CAD in the loop. Furthermore, the simulation expert is able to create simulation experiments based on an early design version while CAD work is still in progress.

The workflow of automated model generation is divided in three main steps. First, existing CAD data has to be processed in CATIA according to a defined structure. Second, after the preparation work an initial package or a model in Modelica is generated and used to setup simulation experiments. Finally, due to variation of design parameter values or manual changes to the CAD part or product the simulation model or package is updated in the loop. In what follows of this section, we will briefly analyze each one of these steps.

### 3.1 CATIA Model Preparation

Starting point is either raw CAD data from other software or existing CATIA parts or products. In terms of package generation the top-level structure of the root product has to be reorganized to match the kinematic structure of the considered mechanism. The generation of connector frames in Modelica is based on CATIA axis systems. Therefore specific geometrical sets with axis systems are created in selected parts of the product structure. These axis systems are either connecting points for joints or other points of interest such as measurement or load points used in simulation. These preparation steps are mandatory for the export of the multibody data. In order to increase usability for the optimization-based de-

sign process, additional design parameters can be defined on the part level. These parameters are either lengths or angles that are linked to certain geometrical constraints of the part. The design parameters change the parts appearance and hence it's mass, inertia and geometrical information. Design parameters need special treatment during the update process as described in the next subsection. The output of the model preparation work can be summarized as follows:

- Top-level product structure equal to kinematic structure,
- Assignment of appropriate materials (densities) to every part of the product structure,
- Connector frames as axis systems in geometrical sets on part level,
- Optional design parameters linked to constraints of the design model on part level.

### 3.2 CATIA Model Conversion to Modelica

Depending on the type of available CAD data different Modelica model structures are produced. CATIA parts (\*.CATPart) are converted to Modelica models and CATIA products (\*.CATProduct) to Modelica packages. Figure 4 shows a CATIA model with two connector frames and the corresponding model structure in Modelica. The Modelica model consists of components from different Modelica libraries parameterized by CAD data. The Modelica MultiBody

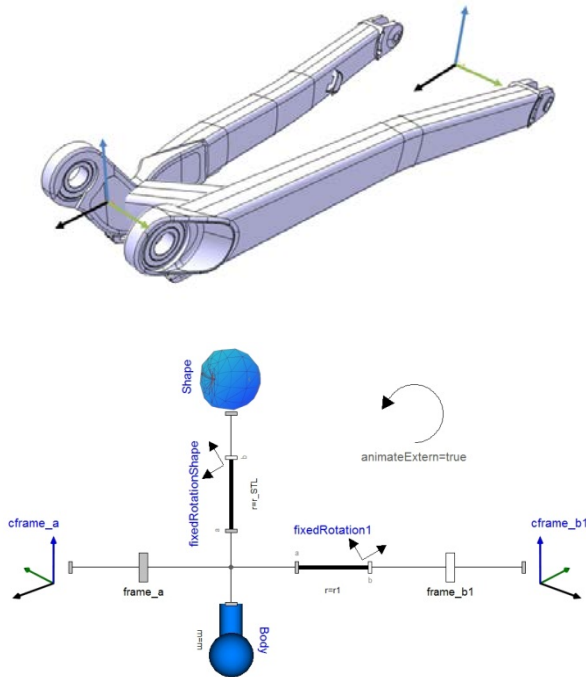


Figure 4: CATIA model (above) and corresponding Modelica model (below)

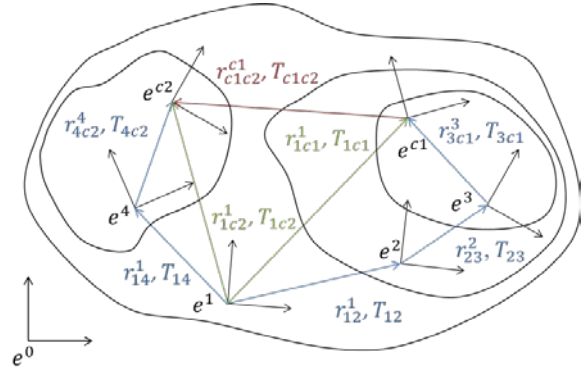


Figure 5: Nested product structure of an assembly in CATIA

Library [5] provides connecting frames (frame\_a, frame\_b1 in Figure 4) that represent the axis systems in the CATIA model. These frames provide cut-forces and cut-torques that are transferred over joints between two components. The first axis system in the connector set of a part or a sub product is set as initial frame\_a for this model. The remaining axis systems are translated to connector frames frame\_b1 to frame\_bn and connected over fixed rotations (fixedRotation) with frame\_a. These models need the relative direction vector and components of the transformation matrix between the frames as input.

Figure 5 illustrates an example of a nested product structure. Every product has its own body fixed axis system  $e^n$  that can be rotated and moved in relation to the axis system on a higher level. CATIA provides the direction vector  $r_{nn+1}^n$  and the transformation matrix  $T_{nn+1}^n$  of every component in the related body fixed axis system  $e^n$ . In this case axis system  $e^{c1}$  is chosen as frame\_a. The inputs for the fixed rotation from frame\_a to frame\_b1 are  $r_{c1c2}^{c1}$  and  $T_{c1c2}$ . Equations (1) and (2) allow a recursive transformation of every axis system to  $e^1$ :

$$\begin{aligned} r_{1c1}^1 &= r_{12}^1 + r_{23}^1 + r_{3c1}^1 \\ &= r_{12}^1 + T_{12}^{-1} r_{23}^2 + T_{12}^{-1} T_{23}^{-1} r_{3c1}^3, \end{aligned} \quad (1)$$

$$T_{1c1} = T_{12} T_{23} T_{3c1}. \quad (2)$$

The relative direction vectors and rotations between  $e^{c1}$  and  $e^{ci}$  in axis system  $e^{c1}$  are calculated with the help of equations (3) and (4):

$$r_{c1c2}^1 = r_{c2}^1 - r_{c1}^1, \quad (3)$$

$$r_{c1c2}^{c1} = T_{1c1} r_{c1c2}^1. \quad (4)$$



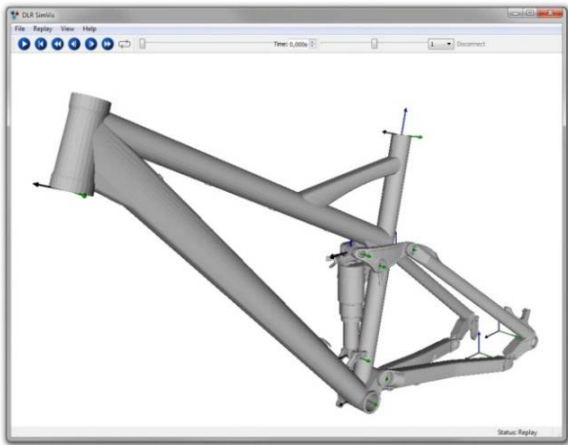


Figure 6: Visualization of a CAD assembly exported from CATIA in DLR SimVis

In addition to the geometric skeleton a body model is introduced containing the mass, a vector to the center of gravity and the body's inertia in respect to the initial frame. If assemblies are translated a model like previously described is generated for every product on top-level that contains connector sets. They are grouped together in a Modelica package.

The visualization is either run internally as Dymola animation with display of the multibody skeleton or externally animated by DLR's SimVis software [6]. SimVis integrates \*.STL (Surface Tessellation Language) files generated from the CAD package as three-dimensional shaded shape representations (Figure 6). Every frame has a visualizer in form of a coordinate system corresponding to the CAD package to simplify simulation work.

Design parameters are integrated as user input parameters on model level. The multibody parameters are either written in the Modelica model as fixed values or stored in a separate textual data file. In the second case the parameters are read into the model over the function `readRealParameter()` to enable parameterized simulation work based on the textual data file.

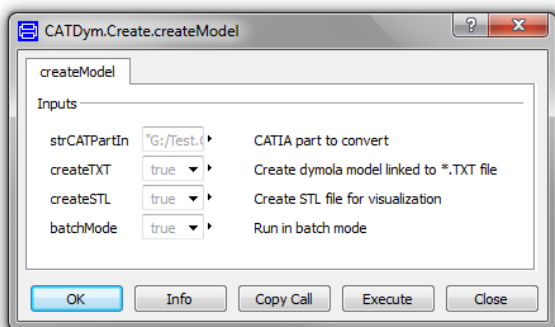


Figure 7: Graphical User Interface for model generation in Dymola

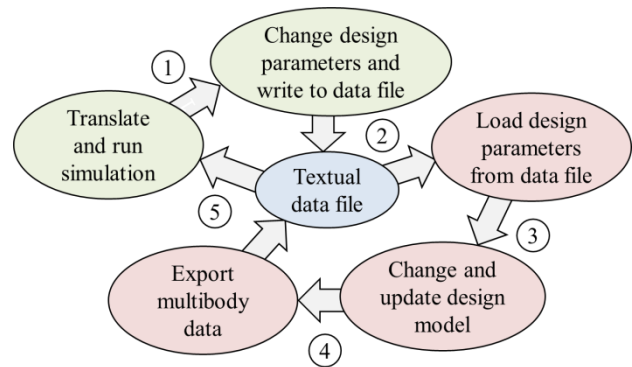


Figure 8: Update process in Dymola (green) and CATIA (red)

The model generation routine is triggered by a Modelica scripting function in Dymola that executes CATIA and runs macros from a VBA Library. Figure 7 shows the Graphical User Interface of the function for model generation with input parameters. The multibody parameters are either stored directly in the model as fixed values or optionally separated from the model in a data file. Furthermore the user can choose to export a three-dimensional shape representation for visualization and activate batch processing. The generated models are manually integrated in simulation experiments by the user in order to investigate the kinematic behavior of the mechanic system. Therefore a multibody structure with joints has to be set up and fitted into testing environments in Dymola.

### 3.3 Updating the Modelica Model Parameters

Throughout the development process changes to the design are applied due to consequences of simulation results or other new requirements. The design model data is either manually modified or changed by design parameters. Isolating the multibody parameters by a text file from the simulation model enables changes to physical and geometrical data without having to reload the model or package in Dymola. Figure 8 leads through the update process and shows the interaction between the different domains. In a first step the design parameter values of the Modelica model are changed in Dymola and written to a textual data file. Then an update routine in the CAD system is triggered by Dymola. The design model is opened in CATIA and the design parameters are read from the data file. After that they are changed in the CATIA design model. The part or assembly is updated in the CAD system and the modified multibody data is exported to the data file analogous to the model generation process in Subsection 3.2.



Figure 9: Bicycle rear suspension in CATIA

Finally, the corresponding Modelica simulation model is updated with the multibody parameters through the `readRealParameter()` function reading the parameter values from the text file into the model.

### 3.4 Optimization Loop

The commercial Optimization Library [7] provides several numerical optimization algorithms for solving different kinds of optimization tasks. Multiple tuner parameters with complex optimization criteria allow automated investigations to support the engineering design process. The goal of the optimization task is to minimize user defined criteria with optional equality and inequality constraints. A detailed view on the used optimization and evaluation algorithms can be found in [7]. In order to use the algorithms of the Function Optimization toolbox, the aforementioned update process is implemented in a Modelica function and fitted to the following defined structure:

```
partial function PartialCriteriaVariables
  input Real tuners[:];
  output Real criteria[:];
end PartialCriteriaVariables;
```

In this criteria function, the optimization tuner parameters are mapped to the design parameters defined in CATIA. Based on the tuner values an update routine in CATIA is triggered, followed by a simulation run in Dymola using the changed multibody data as described in chapter 3.3. Then, the simulation results stored in the corresponding \*.mat file are loaded into workspace through the `readTrajectory()` function. Finally, the values from the loaded

trajectories can be used to formulate the optimization criteria.

## 4 Examples

The following section shows the previously described workflow of model preparation and package generation for an existing bicycle rear suspension from a 2012/13 Fatmodul Ant [8]. After that, the update process is demonstrated in a manual parameter variation followed by an automated optimization utilizing the Optimization Library.

Starting point of the model preparation work is CAD data in \*.STEP (Standard for the Exchange of Product model data) format. At first the model has to be reorganized with respect to the kinematic structure. The rear suspension is designed as linkage driven single pivot. The shock and the swingarm are connected over additional linkages to generate a progressive leverage ratio between wheel travel and shock absorber travel. The shock, the swingarm (Chain stays) and the upper linkage (Rocker) are mounted in the fixed main frame. The seat stays complete the planar kinematic loop. Figure 9 shows the reorganized model structure on top-level in CATIA. In the next step connector frames are attached to the different parts of the assembly representing the connecting joints between the moving parts. Additionally important load points are defined such as the inertial axis system located in the bottom bracket and the rear dropout. For demonstration purposes a design parameter is defined which refers to the wall thickness of the seat stays. At this point the preparation work in CATIA is finished and the remaining steps are executed in Dymola.

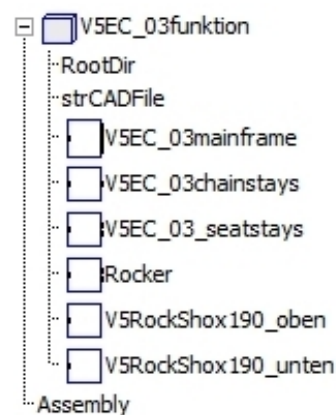


Figure 10: Automatically generated Modelica package of the rear suspension in Dymola

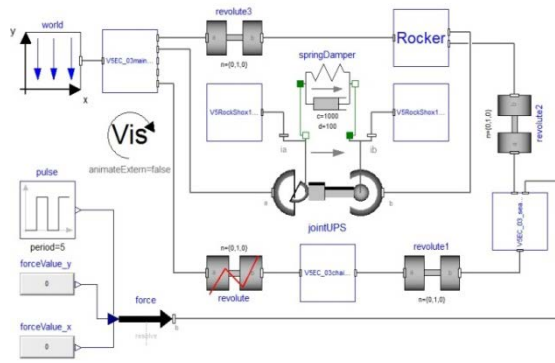


Figure 11: Modelica model of the bicycle rear suspension

A function call in Dymola triggers the package generation in the working directory of the CATIA assembly product. Figure 10 shows the automatically generated Modelica package structure in Dymola. A Modelica counterpart is created for every top-level product from the CATIA design model.

The Modelica package and the associated multibody data file are stored in subfolders of the working directory together with \*.STL visualization data. Based on the kinematic structure a new Modelica model is manually set up by the user to include the generated models. Figure 11 shows the assembled kinematic model of the rear bicycle suspension. The different components of the package are connected via rotational joints with each other. Additional Modelica models complete the overall model. The shock is modeled as `springDamper` and an input force at the rear dropout is implemented.

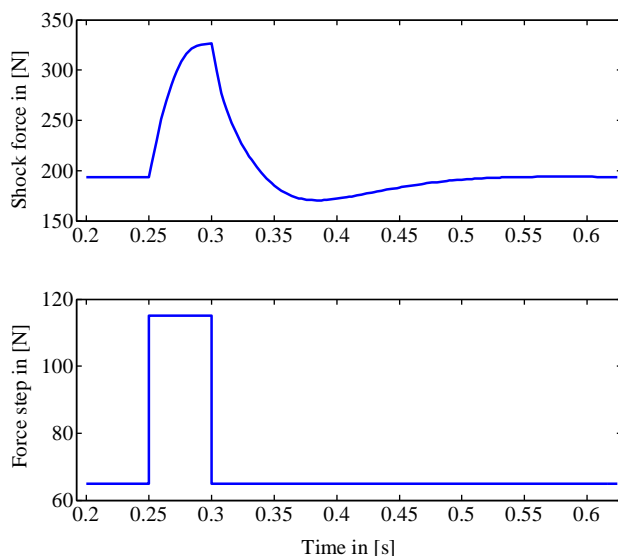


Figure 12: Shock force over time for vertical force step in rear dropout

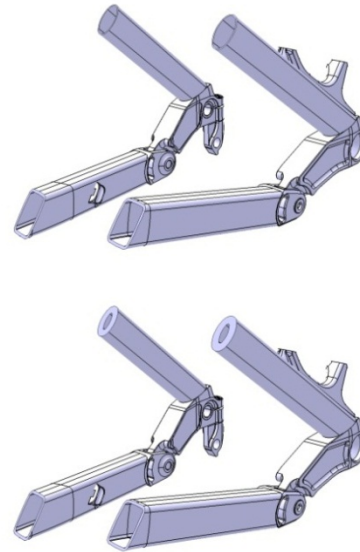


Figure 13: Cut through seat stays with wall thickness  $t = 1$  mm and  $t = 3$  mm

Figure 12 shows the step response of the cut force in shock for a force step along the vertical axis of the rear dropout. An overshooting behavior due to predefined spring and damper parameters can be seen. The CAD model consists of 88 parts in 6 top-level products. The Modelica model for this experiment contains 989 components with 963 time-varying variables formulating 18381 equations. The overall process is executed on a X5650 @ 2.67 GHz Workstation with 12 GB RAM. The execution time for the parameter update process in CATIA together with translation and simulation in Dymola based on an already generated Modelica package is 57 s, which contains 29 s for updating the CAD assembly.

Figure 13 shows the inside of the seat stays with different design parameter values. With the help of the update process the wall thickness of the seat stays is changed between 1mm and 6mm in multiple steps. For every loop of the parameter variation the same routine is executed repetitively.

At first, the current design parameter values are written into the corresponding multibody data file. Then the CAD assembly is opened in CATIA and the design parameter values are read from the previously mentioned file. The parameter values are changed in the CAD product structure and the assembly is updated. In the next step, the updated multibody data, such as mass inertias and geometry, is overwritten in the textual data file by CATIA.

Finally, the model is translated in Dymola in order to reread the changed multibody data from the data file and the simulation experiment is run. The simulation

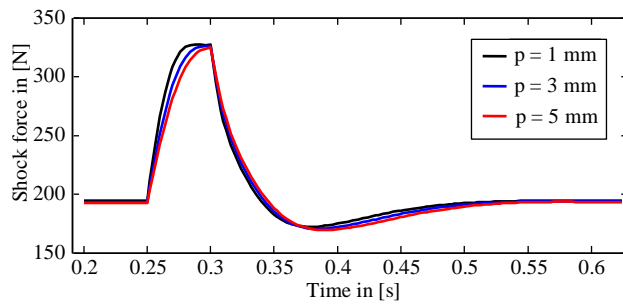


Figure 14: Different shock forces due to variation of wall thickness

results are saved and the next design parameter value is investigated. In this example, no changes to spring stiffness and damping constant were applied during the variation. Figure 14 compares the step responses of the different configurations. Faster response and less overshoot are observed with decreasing wall thickness.

Utilizing the Optimization Library [7], as mentioned in chapter 3.4, allows to investigate the system behavior under complex optimization criteria in comparison to the previously described manual parameter variation. In this example the wall thickness of the seat stays serves as tuner parameter  $p$  and a combined optimization criterion is formulated. The main optimization task is to minimize the overshoot in the shock force, referred to in Figure 15 as  $\Delta f$ . Due to mechanical restrictions in the design of the spring damper system the shock travel  $d$  should not exceed

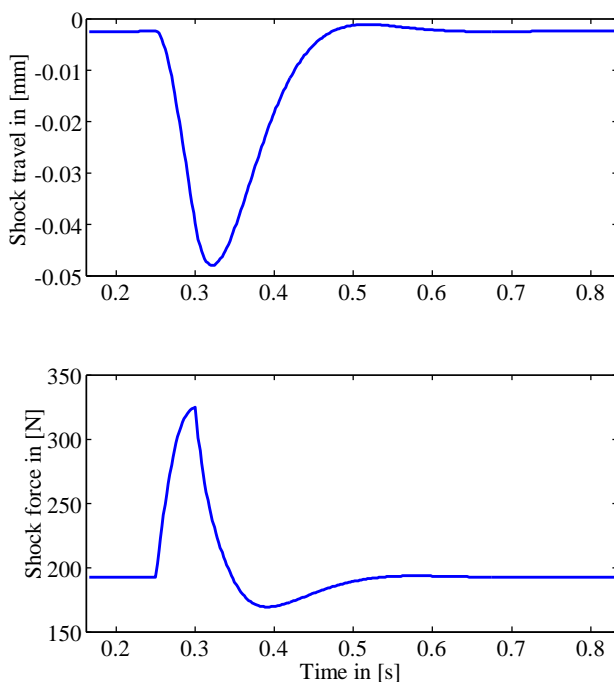


Figure 15: Shock travel  $d$  and overshoot in shock force  $\Delta f$  as optimization criteria

Parameter	Value
Number of criteria evaluations	29
Initial wall thickness $p^0$	3 mm
Range of wall thickness $p$	1 mm – 6 mm
Optimization method	Pattern search
Computation time	1766 s
Best tuner parameter $p^*$	5.43 mm
Overshoot of shock force $\Delta f^*$	23.14726 N
Shock travel $d^*$	47.99986 mm

Table 2: Optimization preferences and results

48 mm. The following formulation summarizes the optimization problem:

$$\min_{1 \leq p \leq 6} \Delta f(p) \quad \text{s.t.} \quad d(p) \leq 48. \quad (5)$$

Table 2 summarizes the results of the optimization run. A solution  $p^*$  is found by a Pattern Search algorithm in less than half of an hour execution time. The solution activates the inequality constraint, i.e.  $d^* \approx 48$  mm and minimizes the overshoot of the shock force.

The shown examples serve mainly for demonstration purpose of the working toolchain and shall illustrate the potential of parameterized design and combined

## 5 Conclusions

An automated model conversion from CATIA to Modelica has been described in this paper. In contrast to existing conversion approaches the paper focuses on the simulation process with Modelica in Dymola. The quick and easy CAD model preparation task can be executed solitary by a simulation expert with basic CATIA experience. The model generation and update toolchain after the model preparation in CATIA is completely controlled out of Dymola. The multibody and design parameters are stored in the model as fixed values or in a separate textual data file. In this way the Modelica models that are derived from CAD are separated from the simulation experiments. Design parameters together with the Optimization Library enable automated parametric design studies. The algorithms and methods in this Library enable the optimization of a part or an assembly in CATIA considering the dynamic behavior of the underlying multibody system.



In the future the focus will be on code enhancement to decrease processing time. Direct communication between Dymola and CATIA via the Active-X-Com interface could improve the model generation process. Further application examples will validate the process quality.

## 6 References

- [1] Modelica Association: "Modelica - A Unified Object-Oriented Language for Systems Modeling", [www.modelica.org](http://www.modelica.org).
- [2] Dassault Systemes AB: CATIA, [www.3ds.com/products-services](http://www.3ds.com/products-services).
- [3] P. Bhattacharya, N. Suyam Welakwe, R. Makanaboyina, A. Chimalakonda, "Integration of CATIA with Modelica," in *The 5th International Modelica Conference, Vienna, Austria, 2006*, pp. 671–675.
- [4] H. Elmqvist, S. E. Mattsson, C. Chapuis, "Redundancies in Multibody Systems and Automatic Coupling of CATIA and Modelica," in *The 7th International Modelica Conference, Como, Italy, 2009*, pp. 551–560.
- [5] M. Otter, H. Elmqvist, "The New Modelica MultiBody Library," in *The 3rd International Modelica Conference, Linköping, Sweden, 2003*, pp. 311–330.
- [6] T. Bellmann, "Interactive Simulations and advanced Visualization with Modelica," in *The 7th International Modelica Conference, Como, Italy, 2009*, pp. 541–550.
- [7] A. Pfeiffer, "Optimization Library for Interactive Multi-Criteria Optimization Tasks," in *The 9th International Modelica Conference, Munich, Germany, 2012*, pp. 669–679.
- [8] Fatmodul Bicycles: [www.fatmodul.de](http://www.fatmodul.de).